

(19)



JAPANESE PATENT OFFICE

## PATENT ABSTRACTS OF JAPAN

(11) Publication number: 10187467 A

(43) Date of publication of application: 21.07.98

(51) Int. Cl.

G06F 9/46  
G06F 13/00  
G06F 15/16

(21) Application number: 08350488

(22) Date of filing: 27.12.96

(71) Applicant: HITACHI LTD HITACHI PROCESS COMPUT ENG INC

(72) Inventor:  
KODAIRA FUMIKO  
ANDO NOBUYOSHI  
HIRASAWA SHIGEKI  
WATAYA HIROSHI  
MATSUZAKI KENJI  
TAMURA SEIICHI  
SUZUKI ATSUSHI

## (54) REMOTE PROCEDURE CALL PROCESSING METHOD

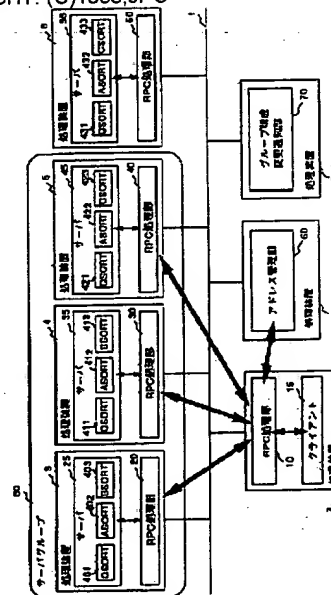
calling all the address information from an address managing part 60.

## (57) Abstract:

COPYRIGHT: (C)1998,JPO

**PROBLEM TO BE SOLVED:** To reduce the change processing of group constitution and to enable early detection concerning a remote procedure call(RPC) processing system for calling a group server.

**SOLUTION:** When adding a server to a server group 80, an additional report is outputted from a group constitution change reporting part 70 to a processor provided with a server in the group such as a processor 5, for example. An RPC processing part 50, which receives this report, holds group constitution change information reporting the addition of new procedure. An RPC request source such as a client 15 of processor 2, for example, issues the RPC request of 'QSORT' to servers 25, 35 and 45 in the server group 80 and correspondent servers in the group are respectively executed. When executing a server 421 and transmitting the executed result to the request source, the processor 5 adds this group constitution change information. When any change in group constitution is detected out of a response message, an RPC processing part 10 at the request source updates its own address information by



(19) 日本国特許庁 (J P)      (12) 公開特許公報 (A)      (11) 特許出願公開番号  
**特開平10-187467**  
 (43) 公開日 平成10年(1998) 7月21日

|                           |       |                              |
|---------------------------|-------|------------------------------|
| (51) Int.Cl. <sup>6</sup> | 識別記号  | F I                          |
| G 0 6 F    9/46           | 3 6 0 | G 0 6 F    9/46      3 6 0 B |
| 13/00                     | 3 5 7 | 13/00      3 5 7 Z           |
| 15/16                     | 3 7 0 | 15/16      3 7 0 N           |

審査請求 未請求 請求項の数 5 O L (全 27 頁)

|           |                  |          |  |
|-----------|------------------|----------|--|
| (21) 出願番号 | 特願平8-350488      | (71) 出願人 | 000005108<br>株式会社日立製作所<br>東京都千代田区神田駿河台四丁目6番地               |
| (22) 出願日  | 平成8年(1996)12月27日 | (71) 出願人 | 000233158<br>日立プロセスコンピュータエンジニアリング株式会社<br>茨城県日立市大みか町5丁目2番1号 |
|           |                  | (72) 発明者 | 小平 富美子<br>茨城県日立市大みか町五丁目2番1号 日立プロセスコンピュータエンジニアリング株式会社内      |
|           |                  | (74) 代理人 | 弁理士 高橋 明夫 (外1名)  |

最終頁に続く

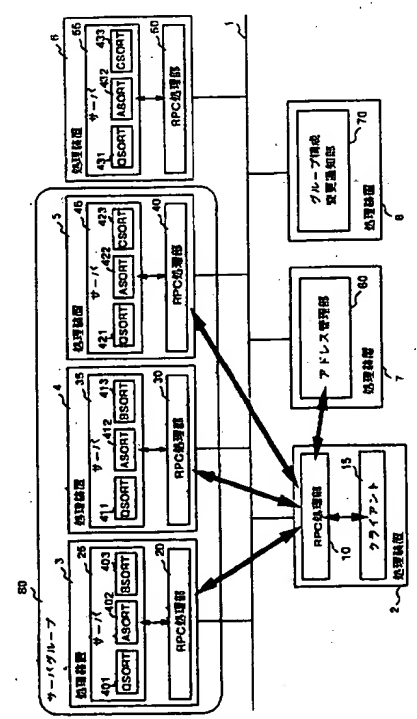
(54) 【発明の名称】 リモートプロシジャコール処理方法

(57) 【要約】

【課題】 グループサーバを呼び出すRPC処理方式において、グループ構成の変更処理を軽減し、早期検出を可能にする。

【解決手段】 サーバグループ80にサーバを追加する場合、グループ構成変更通知部70からグループ内のサーバを具備する処理装置、たとえば処理装置5に追加通知を出す。この通知を受信したRPC処理部50は、新たなプロシジャの追加を通知するグループ構成変更情報を保持する。RPC要求元、たとえば処理装置2のクライアント15は、サーバグループ80のサーバ25, 35, 45へ「QSORT」のRPC要求を発行し、グループ内の対応サーバがそれぞれ実行される。処理装置5はサーバ421を実行して実行結果を要求元に送信する際に、前記グループ構成変更情報を付加する。要求元のRPC処理部10は、応答メッセージからグループ構成の変更を検出すると、アドレス管理部60からすべてのアドレス情報を呼び出し、自らのアドレス情報を更新する。

図 1



## 【特許請求の範囲】

【請求項1】 1つのリモートプロシジャコールの要求に対応し、グループ化された複数のサーバを呼び出すリモートプロシジャコール処理方法において、前記サーバグループの構成の変更の際し、追加または削除する変更サーバの変更情報を、前記サーバグループに所属する任意のサーバが保持し、このサーバが各々のリモートプロシジャコールの要求元から出された実行要求に対応する実行結果を要求元に返すときに、前記実行結果に前記変更情報を付加することを特徴とするリモート

プロシジャコール処理方法。  
【請求項2】 1つのリモートプロシジャコールの要求に対応し、グループ化された複数のサーバを呼び出すリモートプロシジャコール処理方法において、前記サーバグループの任意の1つにグループ構成の変更が通知され、その通知を受け取ったサーバが追加または削除する変更サーバの変更情報を保持し、リモートプロシジャコールの要求元からグループ化された複数のサーバへプロシジャ実行要求が出されると、前記変更情報を保持するサーバは前記実行要求の1つを受け取りそれに対応するプロシジャの実行結果を応答メッセージとして前記要求元に返す際に前記変更情報を付加し、前記要求元は自らが呼び出すサーバグループのメンバを前記変更情報に基づいて変更することを特徴とするリモートプロシジャコール処理方法。

【請求項3】 1つのリモートプロシジャコールの要求に対応し、グループ化された複数のサーバを呼び出すリモートプロシジャコール処理方法において、前記サーバグループの任意の1つにグループ構成の変更が通知され、その通知を受け取ったサーバが自らが保持している変更情報をインクリメントし、一方リモートプロシジャコール要求元は、グループ化された各サーバへ送信するプロシジャの実行要求に、呼び出すサーバ毎に記憶している現在の変更情報を付加して送信し、前記サーバの各々は前記実行要求の1つを受信すると、それに対応するプロシジャの実行を行なうと共に、実行要求に付加されている変更情報と自らの保持する変更情報を比較して不一致のときに、前記要求元に返送する応答メッセージに前記実行結果と共に変更フラグをセットし、前記応答メッセージを受信した要求元は、前記変更フラグがセットされているとき、自分が呼び出すサーバグループのメンバを変更することを特徴とするリモートプロシジャコール処理方法。

【請求項4】 1つのリモートプロシジャコールの要求に対応し、グループ化された複数のサーバを呼び出すにあたり、前記サーバグループのメンバの中からリモートプロシジャコール要求元によって任意に選択した1つのサーバを代表して呼び出し、この呼び出された代表サーバは自らの処理を実行するとともに、前記要求元から要

求された内容を複製して残りのメンバのサーバを呼び出すリモートプロシジャコール処理方法において、前記サーバグループの任意の1つにグループ構成の変更が通知され、その通知を受け取り追加または削除する変更サーバの変更情報を保持しているサーバは、前記要求元から前記代表サーバへプロシジャの実行要求が出され、前記代表サーバから複製の要求内容を転送されたとき、そのプロシジャ実行による実行結果に前記変更情報を付加した応答メッセージを前記代表サーバに返送し、前記代表サーバは、自分が呼び出すサーバグループのメンバを前記変更情報に基づいて変更することを特徴とするリモートプロシジャコール処理方法。

【請求項5】 請求項1～4のいずれか1項において、前記グループ構成の変更として、前記サーバグループに新規サーバを追加し、前記新規サーバの追加を検知したリモートプロシジャコールの各要求元が、このときまで処理要求先としていたサーバと前記新規サーバのいずれかを新たな処理要求先とするか選択し、プロシジャ実行側が行う処理を前記グループ内のサーバ間で分散させることを特徴とするリモートプロシジャコール処理方法。

## 【発明の詳細な説明】

## 【0001】

【発明の属する技術分野】本発明は、分散処理システムで用いられる通信処理技術、とりわけ依頼応答通信の1つであるリモートプロシジャコール（以下、RPCと略称）の処理技術に係わり、特に、サーバをグループした構成のシステムにおいて、サーバグループの構成を変更する方法に関する。

## 【0002】

【従来の技術】RPCは特にクライアント・サーバシステムにおける通信技術として定着し、クライアントからサーバへの1対1の依頼・応答通信を行なうものである。たとえば、複数の情報処理装置がネットワークを介して相互に接続された分散処理システムにおいて、ある処理装置内に存在するクライアントが、自分の処理装置内に存在するプロシジャ呼び出しと同じ手法によって他の処理装置内に存在するサーバ側が提供するプロシジャを呼び出し、その実行結果を受けとる。代表的なRPCとしては、OSF（Open Software Foundation）のDCE（Distributed Computing Environment）での採用例がある。以下では、この例をDCE/RPCと呼ぶ。

【0003】DCE/RPCでは、ディレクトリ・サーバと呼ばれるアプリケーションサーバが、分散処理システム内に存在するすべてのアドレス情報を管理している。クライアントはこのディレクトリ・サーバに問い合わせ呼び出したいサーバのアドレス情報を取得し、RPCによりこのアドレス情報のサーバを呼び出す。DCE/RPCでは、新しいサーバを分散処理システム内に追加する際に、まずディレクトリ・サーバに自分のアドレス情報を登録して、クライアントがサーバのアドレス

を参照できるようにしている。

#### 【0004】

【発明が解決しようとする課題】DCE/RPCなど従来のPRC処理技術では、クライアントはディレクトリ・サーバを検索することでしか、新しいサーバの追加を検出できない。このため、同一処理を行なうサーバをグループ化して多重化サーバとし、クライアントがグループのすべてのサーバにアクセスするような場合（デュアル方式の多重化など）、オンラインでのサーバの追加が困難になる。

【0005】すなわち、オンラインでの追加には、すべてのクライアントが常時ディレクトリ・サーバを検索していることが必要で、ネットワークやディレクトリ・サーバ、さらにはクライアントの負荷が増大してしまうからである。この結果、ディレクトリ・サーバへのアクセススループットが低下すると、サーバグループへのRPCの処理も遅くなる。

【0006】この場合、ディレクトリ・サーバの検索周期を長くして負荷の低減をはかると、タイミングによりあるクライアントは追加サーバを呼び出せ、別のサーバは呼び出せないというように、多重化サーバの構成が不整合となる。たとえば多重化ファイルや多重化データベースが不整合な場合、RPC処理ひいてはシステムの信頼性が著しく低下する。

【0007】サーバの変更をブロードキャスト通信を用いて、即座にすべてのクライアントに伝達する方法が考えられる。このためには、クライアントが常にネットワークからの受信準備をしている必要があり、メモリやマシン負荷が増大する。一般に、クライアント・サーバシステムは、クライアント側に比較的处理能力の低い計算機が用いられることが多く、かかる計算機の負担が大きい。また、ブロードキャストによる送信は必ず受信されるという保証がなく、メッセージ抜けが発生したり、クライアントがサーバの変更情報を受信しない可能性がある。たとえば、クライアントが動作不能状態の場合に変更情報は反映されない。

【0008】本発明の目的は、従来技術の問題点に鑑み、複数のサーバによるサーバグループの構成の変更を、負荷を増大させることなくクライアント側が直ちに検知できるリモートプロシジャコール処理方法あるいは通信方法を提供することにある。

#### 【0009】

【課題を解決するための手段】上記の目的は、1つのリモートプロシジャコールの要求に対応し、グループ化された複数のサーバを呼び出すリモートプロシジャコール処理方法において、前記サーバグループの構成の変更に際し、追加または削除する変更サーバの変更情報を、前記サーバグループに所属する任意のサーバが保持し、このサーバが各々のリモートプロシジャコール要求元から出された実行要求に対応する実行結果を要求元に返すと

きに、前記実行結果に前記変更情報を付加することにより達成される。

【0010】サーバグループの変更はグループの任意の1つに通知され、それを受け取ったサーバが前記変更情報を保持し、前記プロシジャの実行結果を応答メッセージとして前記要求元に返す際に前記変更情報を付加する。一方、前記要求元は自らが呼び出すサーバグループのメンバを前記変更情報に基づいて変更する。

【0011】前記要求元は、予め自らが呼び出すサーバグループのアドレス情報と現在の変更情報を保持していて、前記応答メッセージの変更情報と自分の保持する変更情報を比較し、相違するときに自らのグループ情報を変更する。

【0012】または、リモートプロシジャコール要求元はグループの各サーバに送信する要求メッセージに、各々のプロシジャ実行要求と、保持している現在の変更情報を付加し、それを受け取ったサーバが、プロシジャの実行結果を応答メッセージとして前記要求元に返す際に、自分の保持する変更情報と前記要求メッセージの変更情報が相違するとき、グループ構成変更フラグをセットする。前記要求元は応答メッセージに前記変更フラグがセットされているとき、自分が呼び出すサーバグループのメンバを変更する。

【0013】他の発明として、前記サーバグループのメンバの中からリモートプロシジャコール要求元によって任意に選択した1つのサーバを代表して呼び出し、この呼び出された代表サーバは自らの処理を実行するとともに、前記要求元から要求された内容を複製して残りのメンバのサーバを呼び出すリモートプロシジャコール処理方法において、前記グループ構成の変更の通知を受け取りその変更情報を保持しているサーバは、前記要求元から前記代表サーバへプロシジャの実行要求が出され、前記代表サーバから複製の要求内容を転送されたとき、そのプロシジャ実行による実行結果に前記変更情報を付加した応答メッセージを前記代表サーバに返送し、前記代表サーバは自らが呼び出すサーバグループのメンバを前記変更情報に基づいて変更することを特徴とする。

【0014】代表サーバは、メンバの変更を行なったとき、自らの保持する変更情報を更新し、この変更情報とグループ内の各サーバの実行結果を含む応答メッセージをリモートプロシジャコール要求元に送信する。要求元は、代表サーバからの応答メッセージの変更情報と自らの保持するそれを比較し、相違するときに自らのグループ情報を変更する。

【0015】上記において、前記サーバグループの任意の1つにグループ構成の変更が通知される前に、システム内のサーバのアドレスを管理するネームサーバに対し変更サーバのアドレスを登録または削除し、リモートプロシジャコール要求元あるいは前記代表サーバが前記変更情報を受け取ったときに、前記ネームサーバからアド

レス情報を取得し、自らが呼び出すサーバグループのメンバを変更することを特徴とする。

【0016】このサーバグループのメンバを変更する際に、各サーバが具備するプロシジャの処理内容を識別するための識別子に基づいてグループ構成の制約をチェックし、前記メンバ変更の可否を判定する。たとえば、ある識別子のグループ内での追加または削除に制限がある場合、その制限を越える追加または削除を禁止する。

【0017】上記サーバは、プロシジャとして提供されるプログラムである。あるいは、実行要求に従い自らの処理を実行しその結果を要求元に返す、または結果は返さないプログラムである。

【0018】本発明の構成によれば、クライアントとサーバ間で行なわれるRPC処理に、サーバグループの構成変更を検知して、サーバの変更を自動的に行う機能を持たせているので、従来ユーザが行っていた、クライアントが呼び出し先サーバを変更するために必要な情報の収集処理や、この情報を全クライアントに反映するタイミングの制御といった処理を自動化できる。

【0019】これによれば、多重化サーバグループに既存のサーバと同機能をもつサーバを追加し、多重化サーバの多重度をオンラインで可変することができる。または、前記サーバグループに新規サーバを追加し、前記新規サーバの追加を検知した各々のリモートプロシジャ要求元が、このときまで処理要求先としていたサーバと前記新規サーバのいずれかを新たな処理要求先とするか選択し、プロシジャ実行側が行う処理を前記グループ内のサーバ間で負荷分散させることができる。あるいは、前記サーバグループに新バージョンの新規サーバを追加し、前記新規サーバの追加を検知した各々のリモートプロシジャ実行要求元が、このときまで処理要求先としていたサーバの代わりに前記新規サーバを処理要求先として切り替え、オンラインのバージョンアップを行なうことができる。

#### 【0020】

【発明の実施の形態】以下、本発明の実施の形態を図面にしたがって詳細に説明する。実施例1は、サーバグループの変更を含むRPC処理、実施例2はサーバ側に変更処理を行なわせる代案、実施例3は間接マルチキャスト方式による代案を示す。また、実施例4はサーバグループの変更をRPCに代えて依頼応答通信によって処理する例を示し、各図を通し原則として、同等の要素には同一の符号を付している。

【0021】【実施例1】図1は、本発明の一実施例で、RPCを使用した分散システムの機能ブロック図を示す。処理装置2～8は通信媒体1に接続されて、他の処理装置との間で通信を行う。通信媒体1としては、LAN (Local Area Network) やWAN (WideArea Network)、あるいはWireless (無線)等の通信回線を用いる。バス型を例に示しているが、二重化バス型、一重ル

ープ型、二重ループ型、リング型などの種々のネットワークを用いることができる。まず、RPCを使用した分散システムの構成について説明し、その後グループへの処理装置の追加を例にして、グループ構成の変更処理を説明する。

【0022】処理装置2はクライアントとRPC処理部を具備しており、RPCを用いて他の処理装置に対して処理の実行を要求する。処理装置3～6はサーバとRPC処理部20を具備しており、他の処理装置から要求された処理を実行し、その結果を要求元に返す。処理装置3～5はグループ化されて多重化サーバ80を構成している。

【0023】処理装置2のクライアント15は、ユーザによって作成、あるいは利用されるユーザプログラム、アプリケーションプログラム等であり、処理のために必要な情報の参照を、他の処理装置に提供されているサーバのプロシジャに要求するものである。クライアント15は、サーバのプロシジャの実行を要求する際、RPC要求をRPC処理部10に対して発行する。

【0024】RPC処理部10は、クライアント15のRPC要求処理を代行する。RPC処理部10はクライアント15から発行されたRPC要求を受け、他の処理装置に対し、サーバのプロシジャ実行要求をRPC要求メッセージとして送信し、自らが送信したRPC要求メッセージに対応するプロシジャ実行結果が付されたRPC応答メッセージを受信し、要求元のクライアント15に返す機能を有する。

【0025】処理装置3は、RPC処理部20とサーバ25とを具備しており、他の処理装置から要求された処理を実行し、その結果を要求元に返す。RPC処理部20は、他の処理装置から送信されたRPC要求メッセージを受信し、実行要求されたサーバのプロシジャを実行し、その実行結果を付したRPC応答メッセージを要求元の処理装置に対して送信する。

【0026】サーバ25は、RPC処理部20からの要求に応じて自らを実行し、その実行結果をRPC処理部20に返すプログラムであって、プロシジャの形で提供される。サーバ25は、ユーザにより作成あるいは利用されるプログラムであって、クライアントの処理に必要な情報を提供するプログラムである。

【0027】システム内には、その目的や用途に応じた複数のサーバが種々に組み合わせられて存在する。各々のサーバには、自身がどのような内容の処理を行うかを示す処理識別子が付与され、その処理内容が識別される。処理識別子としては、例えば、サーバが提供するプロシジャのプロシジャ名や、プロシジャを一意に識別するID番号などを用いる。図示のサーバ25は、処理識別子がそれぞれ「Q SORT」のサーバ401、「ASORT」のサーバ402及び「B SORT」のサーバ403が含まれている。

【0028】処理装置4~6も、処理装置3と同様の構成であり、それぞれサーバ35、45、55とRPC処理部30、40、50を具備する。RPC処理部30、40、50は、RPC処理部20と同様の機能を有する。また、サーバ35はサーバ25と同様に、サーバ411~413からなる。サーバ45はサーバ421、422と、識別子が「CSORT」のサーバ423からなる。さらに、グループ外の処理装置6のサーバ55には、サーバ45と同じサーバ431、432及び433が含まれている。

【0029】処理装置7は、アドレス管理部60を具備しており、システム内に存在するサーバを具備している処理装置3~6のアドレスを管理している。アドレス管理部60は、他の処理装置からの要求に応じて、アドレス情報の追加、削除あるいは参照処理を行い、処理結果を要求元に返す。

【0030】処理装置8は、グループ構成変更通知部70を具備しており、ユーザからのコマンド入力等により、サーバグループの構成変更すなわちサーバグループ80へのサーバの追加あるいは削除の要求を受けつけ、そのイベントをサーバグループ80のサーバへ通知する。

【0031】本実施例では、サーバ25、35、45からなるサーバグループ80に対し、サーバ55を追加する変更処理を例に、本発明のリモートプロシジャコール処理方式を説明する。サーバのグループ化には種々の手法があるが、本実施例では同一の処理識別子をもつサーバを1つのグループとして扱う。この場合、同一グループ内のすべてのサーバがまったく同じ処理を行う必要はなく、図1の例のように処理内容が異なってもよい。

【0032】次に、グループ化されたサーバへのRPC処理の概要について説明する。クライアント15は、RPC処理部10を介し、サーバグループ80にプロシジャの実行要求を送信する。ただし、RPC処理部10は、クライアント15が発行したRPC要求の内容から、それを処理できるサーバグループとそれに属するサーバを判定し、そのサーバを具備する処理装置すべてに対してRPC要求メッセージを送信する。この判定は、クライアント15がRPC処理部10に対してRPC要求を発行した際、クライアント15から渡された処理識別子を基に行う。

【0033】サーバグループ80内の処理装置のうち、RPC要求メッセージを受け取った各処理装置のRPC処理部は、要求された処理の実行可能なサーバを実行し、その実行結果をRPC応答メッセージとして要求元の処理装置2に送信する。要求元のRPC処理部10は、RPC要求メッセージを送信した各処理装置からのRPC応答メッセージを受信し、その中から予め決められた論理に基づいて1つのRPC応答メッセージを選択

し、そこに含まれているサーバの実行結果をクライアント15に返す。この選択に用いる論理としては、「ランダムに1つを選択する」、「応答内容を基に多数決論理に基づいて1つを選択する」、「最先着のものを選択する」などの論理を用いることができる。

【0034】次に、各処理装置間で伝送されるRPCメッセージのフォーマットを説明する。図2は、サーバグループの構成変更を通知するRPCメッセージのフォーマットを示す。サーバ変更通知RPCメッセージ100は、グループ構成変更通知部70から送信され、RPC制御ヘッダ部101と変更サーバ処理識別子部103で構成される。RPC制御ヘッダ101にはRPC要求メッセージ、RPC応答メッセージの送受信に伴う処理の制御に必要な情報が格納される。その一部のエリアであるRPC要求内容識別子部102には、RPCで要求される処理内容を受信側のRPC処理部で識別できる処理識別子が格納される。また、送信先アドレス、送信元アドレス、送信通番、要求/応答の識別情報などが格納されるエリアも有している。変更サーバ処理識別子部103には、たとえばサーバグループ80に追加されるサーバ55の処理内容を表わす処理識別子が格納される。

【0035】図3は、RPC要求メッセージのフォーマットを示す。RPC要求メッセージ110は、RPC処理部10がクライアント15のRPC要求に応じて送信するもので、RPC制御ヘッダ部101とクライアントの要求データ部111から構成される。クライアント要求データ部111には、サーバを実行するためのサーバプロシジャの入出力引数が格納される。

【0036】図4は、RPC応答メッセージのフォーマットを示している。RPC応答メッセージ120は、RPC処理部20~50が自処理装置のサーバの実行結果をクライアントへ送信するもので、RPC制御ヘッダ部101、サーバ実行結果データ部122と、グループ構成変更通番部121から構成される。サーバ実行結果データ部122にはサーバの実行結果が格納される。グループ構成変更通番部121は本実施例に特有なもので、サーバの追加などサーバグループの構成変更のたびにインクリメントされるグループ構成変更通番が格納される。

【0037】グループ構成変更通番（以下、変更通番と呼ぶ）は、サーバを具備する各処理装置が独立に管理している変更通番である。1つの処理装置内には、自身が具備するサーバが所属するグループ毎に、対応する1つの変更通番が保持される。サーバグループの変更履歴を管理するために、サーバの追加や削除など、構成変更の種類に対応した種類別の変更通番が用意されてもよい。本実施例では、サーバを追加または削除する通知RPCメッセージ100を受信したとき、後述のようにこの変更通番がインクリメントされる。

【0038】一方、RPC要求側は後述のように、予め

RPC要求を発行する対象サーバグループの全サーバの変更通番を保持している。そして、RPC応答メッセージ120を受信したとき、メッセージに格納された変更通番と保持している変更通番を比較し、変更通番の不一致によってサーバグループの変更を検知する。

【0039】以上、システム構成の概略とメッセージフォーマットを説明した。次に、本実施例のリモートプロシジャコールを実施するシステムの詳細な構成と、各部の処理について説明する。なお、処理装置1〜6が具備するRPC処理部、処理装置7が具備するアドレス管理部60、処理装置8が具備するグループ構成変更通知部70の順に説明する。

【0040】図5は、処理装置の詳細な構成を示す。図示の処理装置2は一例を示したもので、他の処理装置3〜8においても同様な構成としてよい。処理装置2はRPC処理部10と共に、クライアント15とサーバ16の両方を具備している。RPC処理部10は、RPCクライアント処理部11、RPCサーバ処理部12、アドレス管理テーブル13、グループ管理テーブル14を具備している。アドレス管理テーブル13は、各処理装置がサーバグループに属するサーバの情報を格納する。アドレス管理テーブル13は、サーバの処理内容を表す処理識別子と、そのサーバを具備する処理装置のアドレスを格納する。

【0041】なお、処理装置がクライアント処理専用であれば、サーバ16やRPCサーバ処理部12は不要となる。また、処理装置がサーバ処理専用の場合は、クライアント15とRPCクライアント処理部11は不要であり、本実施例の場合にはアドレス管理テーブル13も不要となる。

【0042】また、複数のRPC処理部を設けて、クライアントやサーバからのRPC処理を分担してもよい。この場合、他の処理装置からRPC要求メッセージを送出する際の宛先アドレスは、宛先サーバの処理を担当するRPC処理部のアドレス（例えば、この処理装置のアドレスとこのRPC処理部の受信ポート番号）としてもよい。

【0043】図6に、アドレス管理テーブルの構成を示す。処理識別子部131とアドレス部132とから構成され、各々に格納される情報は、RPC処理部10がアドレス管理部60に要求して取得する。図示は図1における構成の一部を示し、少なくとも3台の処理装置のアドレス情報を格納している。すなわち、「QSORT」で識別されるサーバを有する処理装置が3台で、各アドレスは「133.144.8.159:8080」、「133.144.8.160:8052」及び「133.144.8.161:8031」で与えられている。また、「ASORT」で識別されるサーバを有する処理装置があり、そのアドレス「133.144.9.120:6198」が格納されている。アドレスには、図示のようにIPアドレスと受信ポート番号との組を用いている。

【0044】図7に、グループ管理テーブルの構成を示す。グループ管理テーブル14は、各処理装置がサーバグループの構成情報を格納している。同図のように、処理識別子部141、アドレス部142及びグループ構成変更通番部143から構成され、これら3つの情報が一組になって使用される。処理識別子部141、アドレス部142には、アドレス管理テーブル13と同様に、サーバの処理内容を表す処理識別子とその処理装置のアドレスが格納される。

【0045】グループ構成変更通番部143には、サーバグループの構成の変更を管理する変更通番が格納される。図7(a)は図1のグループ構成による一部を示したもので、「QSORT」で識別されるサーバを有する処理装置が3台あり、各々のグループ構成変更通番はそれぞれ、「2」、「3」、「5」である。また、「ASORT」で識別されるサーバを有する処理装置が少なくとも1台あり、そのグループ構成変更通番は「2」である。ここで、同じQSORTに対する変更通番の値「2」、「3」などは、当該アドレスに対応する処理装置での変更回数を反映している。

【0046】図8は、RPCクライアント処理部の処理を示すフローチャートである。RPCクライアント処理部11は、クライアント15からのRPC要求を代行して処理する。

【0047】まず、クライアント15からRPC要求を受け付ける(S101)。このとき、RPCクライアント処理部11には、クライアント15から処理要求するサーバの処理識別子と、その処理に必要な引数が渡される。

【0048】次に、RPC要求を実行できるサーバを具備する処理装置のアドレスを取得するため、アドレス管理テーブル13を検索するが、この際、最初のRPC要求でアドレス管理テーブル13に何も登録されていなかった場合(S102)、アドレス管理テーブル13の初期化処理を行う(S103)。

【0049】図9は、初期化処理S103を示すフローチャートである。RPCクライアント処理部11はアドレス管理部60に対し、アドレス管理部60に格納された、処理識別子に対応する全てのアドレスの参照を要求するRPC要求メッセージを送信し、応答メッセージに付されたアドレス情報を取得する(S1031)。このアドレス情報をアドレス管理テーブル13に登録する(S1032)と共に、グループ管理テーブル14にも同様にアドレスを登録し、それに対応するグループ構成変更通番に「0」を設定する(S1033)。

【0050】次に、RPCクライアント処理部11は、アドレス管理テーブル13を検索し、RPC要求を実行できるサーバを具備するすべての処理装置のアドレスを取得し(S104)、その処理装置の数をカウントする(S105)。この検索は、クライアント15から渡

された処理識別子をキーにして行われ、アドレステーブル15から、クライアントから渡された処理識別子と同じ処理識別子をもつ全ての処理装置のアドレスを取得する。そして、サーバグループの1つのサーバへ要求メッセージを送信する(S106)。すなわち、RPC要求メッセージ110のクライアント要求データ部111に、サーバ実行に必要な引数を書き込み、S104で取得したアドレスで与えられる処理装置を宛先に、このRPC要求メッセージ110を送信する。

【0051】ところで、S104でアドレスを検索した結果、複数の処理装置のアドレスを取得した場合は、取得したすべての処理装置に対してRPC要求メッセージ110を送信する必要がある。そこで、RPC要求メッセージ110の送信後、メッセージ送信済みの処理装置の数をカウントし(S107)、そのカウント値と取得したサーバグループ内のサーバの数を比較する(S110)。比較の結果、カウント値の方が小さい場合は、S104の処理で取得したアドレスの中から、まだRPC要求メッセージ110を送信していないアドレスを選択し(S111)、RPC要求メッセージ110を送信する。このように、S106からS111までの送信処理を繰り返すことによって、サーバグループに属しRPC要求の処理識別子に対応するすべてのサーバに、RPC要求メッセージ110を送信する。

【0052】以上が、クライアントからのRPC要求の送信処理である。RPC要求メッセージ110を送信すると、RPC要求先のサーバから応答メッセージが返送されてくる。RPCクライアント処理部11は、自らのRPC要求先のサーバからの応答メッセージの受信をチェックし(S108)、以下のように応答受信処理を行う(S109)。

【0053】図10に、応答受信処理(S109)のフローチャートを示す。RPCクライアント処理部11は、応答メッセージ120を受信すると(S201)、応答メッセージ120(図4)からメッセージ送信元のアドレスと、RPC要求内容識別子、グループ構成変更通番を取得する(S202)。そして、自己のグループ管理テーブル14を検索し、応答メッセージに付されていたアドレス及びRPC要求内容識別子の両方とも一致するアドレス情報を検出し、そのアドレス情報の変更通番と応答メッセージ120による変更通番を比較する(S203)。

【0054】比較した結果、両者の値が同じ場合は、サーバグループに変更がなかったと判断し、応答受信処理を終了する。一方、応答メッセージ120による変更通番の値が大きい場合、つまりインクリメントされていた場合は、サーバグループに変更があったと判断し(S204)、応答メッセージ120によるグループ構成変更通番を、グループ管理テーブル14のアドレス情報のグループ構成変更通番部143に格納する(S205)。

【0055】次に、RPCクライアント処理部11はネットワーク1を経由し、処理装置7のアドレス管理部60に、全てのサーバのアドレスを参照するRPC要求メッセージを送信し、応答メッセージに付されたアドレス情報を取得する(S206)。次に、取得したアドレス情報と自分のアドレス管理テーブル13の内容を比較し、前者に残る差分を追加サーバのアドレスとして取得する(S207)。

【0056】ここで、アドレス管理部60から取得したアドレス情報は、処理識別子とそれに対応するアドレスの組の集合である。アドレス管理テーブル13も同様である。比較は両集合の構成要素である組単位に行ない、その差分を求める。グループ管理テーブル14以外にアドレス管理テーブル13を設けるのは、この差分処理を簡単にするためである。なお、サーバを削除する変更の場合は、自分のアドレス管理テーブル13の側に残る差分情報が削除サーバのアドレスとなる。

【0057】次に、本実施例では、S207で取得した追加サーバを具備する処理装置のアドレス情報を、アドレス管理テーブル13に追加する(S208)。さらに、同様にグループ管理テーブル14にも追加し、このとき対応するグループ構成変更通番に初期値「0」を格納する(S209)。

【0058】以上が応答受信処理(S109)であるが、S104で複数のアドレスが取得された場合、RPC要求メッセージは複数個送信されるため、応答メッセージも同じ数だけ返送されてくる。したがって、RPCクライアント処理部11は、すべての応答メッセージを受信するまで、応答受信処理を繰り返し(S112)、複数の応答メッセージに付されたサーバ実行結果データ122の1つを選択し、クライアント15に渡す(S113)。この選択は「ランダムに選択」、「先着で、1番最初に受信したもの」など、予め決められた論理に基づいて行われる。

【0059】次に、RPCサーバ処理部12について説明する。RPCサーバ処理部12はサーバ16のRPC要求受付、および応答返送の代行処理を行うと共に、処理装置7のグループ構成変更通知部70から、サーバグループ構成変更を通知するRPCメッセージを受信し、サーバグループ構成の変更情報を保持する。

【0060】図11は、RPCサーバ処理部の処理を示すフローチャートである。RPCサーバ処理部12は、他の処理装置から送信されたRPC要求メッセージ110を受信すると(S301)、受信メッセージの要求内容を判定する(S302)。この判定は、サーバ変更通知RPCメッセージ100の処理識別子部103を参照することで可能になる。

【0061】受信メッセージがサーバグループの構成変更を通知するメッセージ場合は、自処理装置のグループ管理テーブル14におけるグループ構成変更通番部14



3の値をインクリメントする(S303)。これによって、RPC処理部10はサーバグループの構成変更の情報を保持する。そして、RPC要求元であるグループ構成変更通知部70に対し、グループ構成変更の情報を保持する処理の完了を知らせる応答を返す(S304)。

【0062】一方、ステップS302の判定において、受信した要求メッセージがRPC要求メッセージ110の場合、RPCサーバ処理部12はサーバを実行する

(S305)。すなわち、受信したRPC要求メッセージ110のクライアント要求データ部111からサーバの実行に必要な引数等を取得し、サーバ16に渡すと共に、サーバ16のプロシジャを実行する。処理装置内に同一処理識別子をもつサーバが複数ある場合には、ランダムに1つ選択して実行するものとする。

【0063】サーバ16のプロシジャ実行後、RPCサーバ処理部12はサーバ16から実行結果を受け取り、RPC応答メッセージ120のサーバ実行結果データ部122に格納する。また、グループ構成変更通番部121のエリアには、グループ管理テーブル14のグループ構成変更通番部143の値を格納する(S306)。このRPC応答メッセージ120を、RPC要求元であるクライアントへ送信する(S307)。したがって、このRPC要求時まで、グループ管理テーブル14の変更通番がインクリメントされていれば、この変更通番をRPC応答メッセージに格納して送信し、クライアントにサーバグループの変更を知らせることができる。

【0064】以上、処理装置2~6が具備するRPC処理部の詳細について、クライアントとサーバの両方を備える図5の処理装置の構成で説明した。上記で、図1のシステム構成におけるRPC処理は、クライアント側は処理装置2、グループサーバ側は処理装置3~5に読み代えることで理解できる。

【0065】次に、処理装置7が具備するアドレス管理部60について詳細に説明する。処理装置7のRPC処理部は図示を省略している。アドレス管理部60は、図1に示すシステム内のサーバのアドレスを一括管理しており、ある決められた範囲内(例えば1つのネットワークセグメント内)の処理装置が具備する全てのサーバの処理識別子とその処理装置のアドレスとの対応情報を、自身の管理テーブルに保持している。アドレス管理部60は、他の処理装置からのRPC要求メッセージに応じて、上記情報の追加、削除あるいは参照処理を行い、応答メッセージにより要求元に返す。

【0066】図12は、アドレス管理部の処理を示すフローチャートである。まず、RPC処理要求を受信すると(S401)、その処理要求内容を判定し(S402)、サーバアドレスの追加、サーバアドレスの削除またはサーバアドレスの検索の何れかを処理する。

【0067】サーバの追加時には、他の処理装置の要求元、例えばあるクライアントがアドレス管理部60に対

して、追加サーバを具備する処理装置のアドレスと追加サーバの処理識別子を付したRPC要求メッセージを送信する。アドレス管理部60は要求内容がサーバアドレスの追加と判定すると、処理装置7が具備する管理テーブルに、RPC要求メッセージに付されていたアドレスと処理識別子を追加し(S403)、追加処理の成否を応答メッセージとして要求元に送信する(S404)。

【0068】アドレス管理部60は同一の処理識別子をもつサーバを1つのサーバグループとして管理する。したがって、アドレスとサーバが属するサーバグループとを管理するため、アドレスと処理識別子を対応付けて追加する。

【0069】次に、サーバの削除時には、他の処理装置内の要求元が、削除するアドレスとサーバの処理識別子を付したRPC要求メッセージを送信する。アドレス管理部60は削除要求であると判定すると(S402)、RPC要求メッセージに付された処理識別子とアドレスの組みと同一のアドレス情報を、自身が具備するテーブルから削除し(S405)、削除処理の成否を応答メッセージとして要求元へ送信する(S406)。

【0070】次に、ある処理の実行可能なサーバを有する処理装置のアドレスを参照する際には、他の処理装置内の要求元がその処理の内容を示す処理識別子を付したRPC要求メッセージを、アドレス管理部60に送信する。要求内容がアドレスの参照と判定されると(S402)、アドレス管理部60が具備する管理テーブルを検索し、RPC要求メッセージに付された処理識別子と同じ処理識別子に対応付けられたアドレスを取り出し(S407)、RPC応答メッセージに付して要求元に送信する(S408)。

【0071】ステップS407の検索によって、複数のアドレスが取得された場合は1つのアドレスをランダムに選択してもよいし、すべてのアドレスをRPC応答メッセージに付して送信してもよい。

【0072】次に、処理装置8が具備するグループ構成変更通知部70について説明する。処理装置8のRPC処理部は図示を省略している。グループ構成変更通知部70は、ユーザからのコマンド入力等により指定のサーバグループの構成変更の要求を受け付ける。たとえば、コマンドによりサーバグループ80の構成変更の要求が行なわれ、このコマンドの引数として、追加または削除されるサーバを具備する処理装置のアドレスや、サーバの処理識別子などの情報が、グループ構成変更通知部70に与えられる。

【0073】グループ構成変更通知部70は、このコマンドを受け付けると、処理装置7のアドレス管理部60に対して、アドレスの変更を要求するRPC要求メッセージを送信する。たとえば、新規サーバを具備する処理装置のアドレスの追加を要求するRPC要求メッセージを送信し、アドレス管理部60にアドレスを登録すると

共に、サーバグループ構成変更を通知するRPCメッセージを、指定のサーバグループ80に送信する。サーバグループ80に属する処理装置3〜5は、このサーバグループ構成変更を通知するRPCメッセージを受信し、自身のアドレス管理テーブル13及びグループ管理テーブル14に、サーバグループ構成の変更情報を保持する。

【0074】図13は、グループ構成変更通知部によるサーバ追加処理を示すフローチャートである。グループ構成変更通知部70は、サーバグループへのサーバの追加要求を受け取ると(S501)、アドレス管理部60に、追加対象となるサーバグループに属するサーバのアドレス参照を要求するRPC要求メッセージを送信し、アドレスを取得する(S502)。このとき、グループに属する任意の1つのサーバのアドレスを取得してもよいし、すべてのサーバのアドレスを取得してもよい。後者の場合、グループ構成変更通知部70が任意の1つを選択する。

【0075】次に、グループ構成変更通知部70は、追加サーバのアドレスの登録を要求するRPC要求メッセージをアドレス管理部60に対して送信し、管理テーブルへのアドレス登録を依頼する(S503)。次に、ステップS502で取得したアドレスのサーバへ、サーバの追加を通知するRPCメッセージ100を送信し(S504)、そのサーバから追加情報を記憶する処理が完了したという応答を受信する(S505)。

【0076】以上によって、本発明のリモートプロシジャコール処理を実現する、本実施例の基本的な構成と動作が明らかになった。次に、具体的なリモートプロシジャコールについて、グループサーバに対して新規サーバを追加する処理の例を説明する。この処理の全体的な流れは、サーバグループに対してサーバの追加を通知する処理、各クライアント側処理装置における追加サーバの検知処理に大別でき、この2つに分けて説明する。

【0077】図14は、図1のシステムのRPC処理部を詳細にした構成図で、新規サーバを追加する処理の流れを矢線で示している。また、図15は、クライアント側とサーバ側の各々のグループ管理テーブルの具体的な格納内容を示す説明図である。さらに、図16は、本実施例によって新規サーバを追加するときの全体的な処理を示すフローチャートである。これらの図を参照しながら、本実施例によるサーバの追加処理を説明する。

【0078】グループ構成変更通知部70は、ユーザから、サーバグループ80へのサーバ55の追加を受け取ると、新規サーバ55を具備する処理装置6のアドレスをアドレス管理部60へ登録する。また、グループ構成変更通知部70は、アドレス管理部60からサーバグループ80内の任意の1つのサーバのアドレス、ここではサーバ45のアドレスを取得し、そのアドレスであるRPCサーバ処理部42宛に、矢印201aのようにサー

バ追加の通知RPCメッセージ100を送信する(S601)。

【0079】RPCサーバ処理部42は、サーバ追加の通知RPCメッセージを受信すると、RPCサーバ処理部42内にあるグループ管理テーブル44のグループ構成変更通番を5⇒6にインクリメントし、グループ構成変更通知部70に処理完了の応答メッセージを送信する(S602)。

【0080】次に、クライアント側の処理装置における追加サーバの検知処理について説明する。図17に、クライアント側における追加サーバの検知処理のフローチャートを示す。

【0081】クライアント15は、たとえば処理識別子「QSORT」を含むRPC要求メッセージ110をRPCクライアント処理部11に出す。RPCクライアント処理部11は、アドレス管理テーブル13を検索し、サーバグループ80に属する「QSORT」の実行可能なサーバを具備する処理装置3、4、5のアドレスを取得し、矢印205a、b、cのようにRPC要求メッセージを送信する(S701)。このとき、グループ管理テーブル14の「QSORT」をもつ各アドレスに対応する変更通番は、図15のように記憶されている。

【0082】RPC要求メッセージを受信した処理装置3、4、5のRPCサーバ処理部22、32、42は、RPC要求に応じて矢印206a、b、cのようにサーバを実行し(S702)、サーバの実行結果を矢印207a、b、cと取得する。また、グループ管理テーブル24、34、44内の変更通番を矢印208a、b、cのように取得し、実行結果と変更通番を格納したRPC応答メッセージを矢印209a、b、cのように送信する(S703)。このとき、グループ管理テーブル24、34、44の「QSORT」の変更通番は図15の内容である。

【0083】RPCクライアント処理部11は、処理装置3、4、5からのRPC応答メッセージを受信し(S704)、RPC応答メッセージ内の変更通番とグループ管理テーブル14内の変更通番を比較し、グループサーバの構成が変更されたか否かを判定する(S705)。この処理は、図10の応答受信処理のステップS204に相当する。

【0084】ここでは、処理装置5が具備するRPCサーバ処理部42は、上記のようにサーバ追加の通知RPCメッセージを受信し、グループ管理テーブル44の「QSORT」のグループ構成変更通番を5⇒6にインクリメントし、応答メッセージに付加して矢印209cのように送信している。従って、RPCクライアント処理部11が、RPC応答メッセージの変更通番と、自分のグループ管理テーブル13の同一アドレスの変更通番を比較すると、新規サーバの追加が検知できる。

【0085】新規サーバの追加を検知したRPCクライ

アント処理部 11 は、アドレス管理部 60 からすべてのサーバのアドレス情報を取得し、自分のアドレス管理テーブル 13 のアドレス情報と比較し、差分として得られるサーバ 55 のアドレス情報を、アドレス管理テーブル 13、グループ管理テーブル 14 に追加する (S706)。次に、RPC クライアント処理部 11 は、処理装置 3、4、5 の各々から受信した RPC 応答メッセージの 1 つを選択し、クライアント 15 にサーバの実行結果として渡す (S707)。

【0086】なお、上記ではグループ構成変更通知部 70 が処理装置 8 によって具備される構成としているが、どの処理装置が具備していてもよい。例えば、処理装置 6 がグループ構成変更通知部 70 を具備していてもよい。このとき、サーバ 55 が自身の追加要求をグループ構成変更通知部 70 へ通知し、グループ構成変更通知部 70 へ追加処理要求を依頼するようにしてもよい。

【0087】また、図 11 のステップ S302 の判定がサーバの追加要求の場合、指定のサーバグループに新規のサーバの追加の可否を判定してから、ステップ S303 で追加の処理をするようにしてもよい。

【0088】たとえば、図 1 のサーバグループ 80 において、グループ 80 内で多重化できないサーバがあり、その処理識別子が「CSORT」であるとする。この場合に、サーバ 55 を追加すると、「CSORT」のサーバが多重化されてしまう。従って、RPC サーバ処理部 42 は、RPC 通知メッセージ 100 の追加サーバの処理識別子を受け取ると、グループ管理テーブル 14 の処理識別子と対照し、「CSORT」サーバが多重化される場合には追加処理を中断する。あるいは、サーバグループが完全多重化の構成をとる場合は、グループ管理テーブル 14 の処理識別子と対照し、追加するサーバの処理識別子が同一であるか判定し、同一でなければ追加不可の応答を返す。

【0089】このほかにも、追加処理が許されるユーザ ID やプログラム ID や認証情報を予め RPC サーバ処理部内に記憶しておき、RPC 通知メッセージに要求元のユーザ ID、プログラム ID または認証情報を付加して送信し、受信時にこれらの情報を比較して、追加の可否を判定するようにしてもよい。

【0090】以上、本実施例の構成によれば、RPC を用いて 1 つの要求に対応して複数のサーバを並列的に呼び出すクライアント側の計算機は、各々が独立して非同期的に、複数のリモートプロシジャが構成するグループの構成変化を検出し、グループ構成変更処理を行う。また、この検出のための情報は、通常時の RPC 応答に付加されてクライアント側に伝達される。したがって、グループ構成変更の際に、これに伴う追加処理メッセージがネットワーク上に集中することがなく、ネットワークの負荷、構成変更に伴い追加されたリモートプロシジャ側の計算機あるいはリモートプロシジャのアドレスを

管理するネームサーバ等の負荷を時間的に分散できる。特に、クライアント側計算機がグループ構成の変更の有無を常に監視している必要がなく、自らの RPC 要求時に対する応答内容から変更を検知できるので、クライアント側計算機の変更処理に伴う負荷の増大は少ない。

【0091】また、本実施例によれば、リモートプロシジャ側が多重化サーバを構成し、クライアントが多重化サーバにアクセスしているときに、サーバ追加の前後で各サーバにアクセスするクライアント側計算機を同じにすることができ、グループ構成の変更を行なうことができるので、サーバの内部状態の整合性が保証され、多重化サーバの通信方式として有用である。さらに、サーバの多重度をオンラインに変化させることができる。

【0092】本実施例では、クライアント側はグループ内の全てのサーバに対して RPC 要求を発行している。しかし、グループ内の一部のサーバに RPC を発行するようにしてもよい。これによれば、クライアントが追加サーバを検知したとき、それまで RPC を発行していた要求先のサーバに代えて追加サーバを用い、サーバグループ内で処理の負荷分散を図ることができる。あるいは、要求先の旧バージョンに代えて新バージョンの追加サーバを用い、サーバのバージョンアップをオンラインで行うこともできる。

【0093】また、サーバはプロシジャの形で提供されていなくてもよい。要求に対し自らの処理を実行し、その実行結果を依頼元に返すプログラムであってもよい。すなわち、実行結果を依頼元に返すタイミングが、必ずしも自らの処理の実行終了時である必要はなく、自らの処理の実行途中に何らかのデータを依頼元に返すプログラムであってもよい。さらに、サーバが処理要求に応じて実行するプログラムは、必ずしも処理結果を出力するものである必要はない。処理終了時あるいは処理開始時等に、サーバ側の RPC 処理部が実行結果データ部にデータを何も格納しないで、RPC 応答メッセージを返すようにしてもよい。

【0094】さらに、本実施例では同一の処理識別子は同一の処理内容を表すものとして、図 1 のように多重化サーバのグループ構成を示したが、これに限られるものではない。RPC ではサーバが保持するプロシジャの名前や処理内容に関わらず、異なる名前や処理を行なうプロシジャをもつサーバであっても、同一の処理識別子をもつことが可能である。このような識別子の付与を行なった場合、送信側はグループ内の各サーバが保持するプロシジャの内容を意識しないで送信メッセージを送ることができるので、送信側の負荷低減をはかれる等のメリットがある。

【0095】【第 2 の実施例】本実施例はグループ構成の変更の有無の検出を、第 1 の実施例とは異なりサーバ側で行なう。すなわち、クライアントがサーバへ RPC を発行する際に、RPC 要求メッセージにクライアント

が保持している変更通番を格納して送出し、サーバ側がその変更通番と、自処理装置のグループ管理テーブルに記憶している変更通番を比較して、グループ構成の変更の有無を検出する。以下、第1の実施例と異なる部分を中心に説明する。なお、システム構成は第1の実施例と同じであり、以下では図14の構成を参照する。

【0096】図18に、クライアント側の処理装置が送出するRPC要求メッセージのフォーマットを示す。第1の実施例におけるRPC要求メッセージ(図3)に、クライアントが保持している変更通番を格納する、グループ構成変更通番部112を追加している。

【0097】図19に、サーバ側の処理装置がクライアント側へ実行結果を返すRPC応答メッセージのフォーマットを示す。第1の実施例のRPC応答メッセージ(図4)に、サーバグループにおける構成変更の有無を知らせる変更フラグ部123を追加している。

【0098】RPCクライアント処理部11が行う処理について説明する。まず、第1の実施例による処理フロー(図8)のステップS106において、RPCクライアント処理部11がサーバへRPC要求メッセージを発行する際に、要求先サーバのアドレスに対応する変更通番をグループ管理テーブル14から取り出し、図18(b)のように、RPC要求メッセージのグループ構成変更通番部112に格納する。

【0099】図20は、本実施例によるRPCクライアント処理部の応答受信処理のフローチャートである。RPCクライアント処理部11は、サーバからRPC応答メッセージ120を受信すると(S801)、変更フラグ部123を調べ、サーバグループに追加があったか否かを判定する(S802)。追加フラグがセット(ON)されていれば、サーバグループ80に変更があったと判断し、S803～S807の処理を行う。ここで、ステップS803～S807の処理は、第1の実施例の図10に示したステップS205～S209の処理と同様である。

【0100】次に、サーバ側のRPCサーバ処理部が行う処理について説明する。図21は、本実施例によるRPCサーバ処理部の応答受信処理のフローチャートで、例えば処理装置5のRPCサーバ処理部42によって実行される。同図で、ステップS901～S904は、第1の実施例(図11)におけるS201～S204と同様の処理となる。

【0101】ステップS902の判定で、要求メッセージがサーバの変更でない場合、まずクライアントから要求されたサーバを実行する(S905)。そのサーバから実行結果を受け取ると、RPC要求メッセージ120に付加されていた変更通番を取得し(S906)、自処理装置5のグループ管理テーブル44に管理している変更通番と比較する(S907)。比較の結果、自分の変更通番の方が大きければ、図19(b)のようにRPC

応答メッセージのサーバ追加フラグをONにし(S910)、RPC応答メッセージ120に自分の変更通番を付加して(S911)、サーバの実行結果と共にクライアント側の処理装置2へ返送する(S912)。

【0102】一方、取得した変更通番が自分の変更通番以下であれば、RPC応答メッセージのサーバ追加フラグをOFFにし(S909)、RPC応答メッセージ120に自分の変更通番を付加し(S911)、サーバの実行結果と共に要求元の処理装置へ返送する。

【0103】この後、RPCクライアント処理部11は、応答メッセージ120の追加フラグのON/OFFを判定し、ステップS803～S807により、グループ管理テーブル14とアドレス管理テーブル13を更新する。

【0104】なお、上記ではサーバ追加の例を説明したが、サーバ削除の場合はサーバ削除フラグを用意しそのON/OFFを判定する。あるいは、サーバの追加、削除に共用する変更フラグを用意し、ステップS907における比較の結果、取得したグループ構成変更通番が自分の保持する変更通番と相違する場合に、グループ構成に変更があると判定して、ステップS803～S807の処理を行なうようにしてもよい。

【0105】本実施例によれば、サーバグループ構成変更の有無の判定をサーバ側で行っているため、クライアントの処理が軽減できる。特に、不正なサーバの変更要求が連続するような場合に、クライアントのオーバーロードを回避できる。

【0106】〔第3の実施例〕次に、第3の実施例について説明する。第1、第2の実施例では、クライアントの処理装置がサーバグループのサーバを呼び出す際に、複数のサーバに対して直接RPC要求メッセージを送信するマルチキャスト方式を用いていた。これに対し、本実施例では、クライアントの処理装置はサーバグループの任意のサーバに対してRPC要求メッセージを送信し、このメッセージを受信した処理装置が宛先であるサーバを実行するとともに、グループの残りのすべてのサーバに対してRPC要求メッセージを転送し、その転送メッセージを受信した各処理装置が自身のサーバを実行する、という間接マルチキャスト方式を用いる。

【0107】図22は、本実施例によるサーバ追加を説明するためのシステム構成図で、詳細は図14と同様になる。図23は、本実施例によるRPC要求メッセージと応答メッセージのフォーマットを示す。同図(a)のRPC要求メッセージでは、複製の転送による場合を区別するために、図3のRPC要求メッセージに、転送メッセージフラグ部113を追加している。また、同図(b)の応答メッセージでは、変更フラグ部123を設け、グループ構成変更通番部121は削除している。

【0108】次に、本実施例でのサーバ追加の処理を説明する。この処理の全体的な流れは、サーバグループに

対してサーバの追加を通知する処理と、各々のクライアント側処理装置における追加サーバの検知処理に分けることができる。前者の処理は第1の実施例と同様であるので、以下では後者の処理のみを説明する。

【0109】RPCクライアント処理部11は、グループ内のマスタサーバとして予め設定されている任意の1つ、ここでは処理装置3のサーバ25に対してRPC要求メッセージを送信する。なお、マスターとなる処理装置3のグループ管理テーブル24には、グループ80のすべてのサーバを呼び出せるように、そのアドレスと変更通番を管理している。

【0110】クライアントからのRPC要求メッセージを受け取った処理装置3のRPCサーバ処理部22は、転送メッセージフラグがOFFとなっていることから、転送メッセージでないと判断すると、グループ内の他のすべてのサーバ宛てに転送RPC要求メッセージを送信し、送信された転送RPC要求メッセージは、それぞれ処理装置4、5によって受信される。

【0111】RPC要求メッセージを受信した処理装置4、5内のRPCサーバ処理部32、42は、RPC要求に応じてサーバを実行し、サーバの実行結果を取得し、また、グループ管理テーブル24、34、44内の変更通番を取得し、それらを格納したRPC応答メッセージを、転送RPC要求メッセージを送出したRPCサーバ処理部22へ送信する。

【0112】RPCサーバ処理部22は、自分にグループ構成変更通知のあるときはグループ管理テーブル24を更新する。また、処理装置4、5からのRPC応答メッセージ内の変更通番とグループ管理テーブル24内の変更通番を比較し、グループ構成に変更が有るか否かを判定する。変更がある場合には、グループ管理テーブル24を更新する。

【0113】RPCサーバ処理部20は、クライアントに自分のサーバ25の処理結果と他の転送先サーバ35、45からの実行結果を送信する際、先にグループ管理テーブル24を更新している場合は、変更フラグ部123に変更フラグをセット(ON)する。

【0114】処理装置3からRPC応答メッセージを受信したRPCクライアント処理部11は、メッセージの変更フラグがONであればサーバグループ80の構成に変更があると判断し、アドレス管理部60からすべてのサーバのアドレスを取得し、アドレス管理テーブル13のアドレスと比較する。その結果、差分となるサーバ55のアドレスを、アドレス管理テーブル13およびグループ管理テーブル14に追加する。次に、RPCクライアント処理部11は、RPC応答メッセージ内の処理装置3、4、5のサーバ実行結果から1つを選択し、クライアント15に渡す。なお、本実施例では、クライアント側にはグループ管理テーブルが不要となる。

【0115】本実施例によれば、サーバグループにRP

C要求を行なう場合に、グループ内で任意に選択した処理装置との間でのみ送受信が行なわれ、グループ内の他の処理装置の残りのすべてのサーバに対してRPC要求メッセージを転送して処理する間接マルチキャスト方式を用いるので、クライアントやネットワークの処理付加を低減できる。

【0116】〔第4の実施例〕次に、第4の実施例について説明する。第1、第2、第3の実施例では、クライアントとサーバとの通信方法としてRPCを用いた。RPCは複数の情報処理装置がネットワークを介して相互に接続された分散処理システムにおいて、ある処理装置内に存在するクライアントが、自分の処理装置内に存在するプロシジャ呼び出しと同じ手法によって他の処理装置内に存在するサーバ側が提供するプロシジャを呼び出し、その実行結果を受け取る処理を行う依頼応答通信方法である。しかしながら、RPCは内部的には、(1)クライアントからサーバへの依頼メッセージの通信、(2)サーバからクライアントへの応答メッセージの通信を組合せ、その上で、(3)サーバ側の実行可能なプログラムはプロシジャの形で提供され、(4)クライアントが、自分の処理装置内に存在するプロシジャ呼び出しと同じ手法によって他の処理装置内に存在するサーバ側が提供するプロシジャを呼び出すことができるように処理することを特徴とした、依頼応答通信の1つの形態である。

【0117】本発明の主旨は、クライアントとサーバ間の通信方法として、特にRPCに限らなくとも、クライアントとサーバ間の通信方法が、前記リモートプロシジャコール要求/応答の代わりに、それぞれ少なくとも(1)クライアントからサーバへの実行依頼メッセージの通信、(2)サーバからクライアントへの応答メッセージの通信を組み合わせた依頼応答通信であれば適用可能である。

【0118】第1、第2、第3のいずれの実施例においても、リモートプロシジャコール要求の代わりに依頼メッセージ、リモートプロシジャコール応答の代わりに応答メッセージを用いてもよい。この場合、第1、第2、第3の各実施例におけるシステム構成及び処理装置内の構成は、第4の実施例においても同様であり、処理内容についても同様でよい。また、RPCメッセージのフォーマットについても、RPC制御ヘッダ部を依頼応答通信のための制御情報を含む、依頼応答通信制御ヘッダ部に変更するだけでよい。

【0119】以上のように、本発明の主旨は、クライアントとサーバとの通信方法を特にRPCに限らなくとも、クライアントとサーバとの通信方法が依頼応答通信であれば適用可能であり、RPCの場合と同様の効果を得ることができる。

【0120】

【発明の効果】本発明によれば、クライアント側の計算

機は動作中にサーバグループの構成変化を検知することができる。従って、システム管理者はアプリケーションシステムを稼働させながら、サーバグループにサーバを追加し、新たなサービスを追加したり、多重度を増やしたり、サーバのバージョンアップを行ったりすることができ、システムの拡張性、保守性、耐故障性を向上させることができる。また、アプリケーションシステムを稼働させながら、サーバグループからサーバを削除し、不要サーバを切り離すことができ、システムの縮小性、保守性を向上させることができる。

【0121】また、グループの構成変化の検出情報は、RPC応答または依頼応答に付加されて送受信されるので、変更に伴うネットワーク上の負荷集中を生じない。さらに、クライアント側計算機がグループ構成の変更の有無を常に監視している必要はなく、自らのRPC要求または依頼通信時に対する応答内容から変更を検知できるので、クライアント側計算機の変更処理に伴う負荷の増大はない。

【0122】また、本発明によれば、リモートプロシージャ側が多重化サーバを構成している場合、サーバ追加の前後で各サーバにアクセスするクライアント側計算機を同じにすることができ、サーバの内部状態の整合性が保証されるので、多重化サーバの通信方式として有用である。

#### 【図面の簡単な説明】

【図1】本発明のリモートプロシージャコール処理方式を適用するシステム構成の一例を示すブロック図。

【図2】一実施例によるサーバ変更通知RPCメッセージのフォーマット図。

【図3】一実施例によるRPC要求メッセージのフォーマット図。

【図4】一実施例によるRPC応答メッセージのフォーマット図。

【図5】クライアントとサーバを併設する処理装置の構成図。

【図6】アドレス管理テーブルの内容を示すフォーマット図。

【図7】グループ管理テーブルの内容を示すフォーマット図。

【図8】第1の実施例によるRPCクライアント処理部の全体的な処理動作を示すフローチャート。

【図9】テーブル初期化処理の詳細を示すフローチャート。

【図10】RPCクライアント処理部の応答受信処理を

示すフローチャート。

【図11】第1の実施例によるRPCサーバ処理部の処理動作を示すフローチャート。

【図12】アドレス管理部の処理動作を示すフローチャート。

【図13】グループ構成変更通知部の処理動作を示すフローチャート。

【図14】サーバ追加処理の流れを説明するシステム構成図。

10 【図15】図14のシステム構成において、各グループ管理テーブルの具体的な管理内容を示す説明図。

【図16】サーバ側でのサーバ追加処理の流れを説明するフローチャート。

【図17】クライアント側でのサーバ追加の検知処理を示すフローチャート。

【図18】第2の実施例におけるRPC要求メッセージを示すフォーマット図。

【図19】第2の実施例におけるRPC応答メッセージを示すフォーマット図。

20 【図20】第2の実施例におけるRPCクライアント処理部の応答受信処理の処理動作を示すフローチャート。

【図21】第2の実施例におけるRPCサーバ処理部の処理動作を示すフローチャート。

【図22】第3の実施例における間接マルチキャスト方式の処理の流れを説明するシステム構成図。

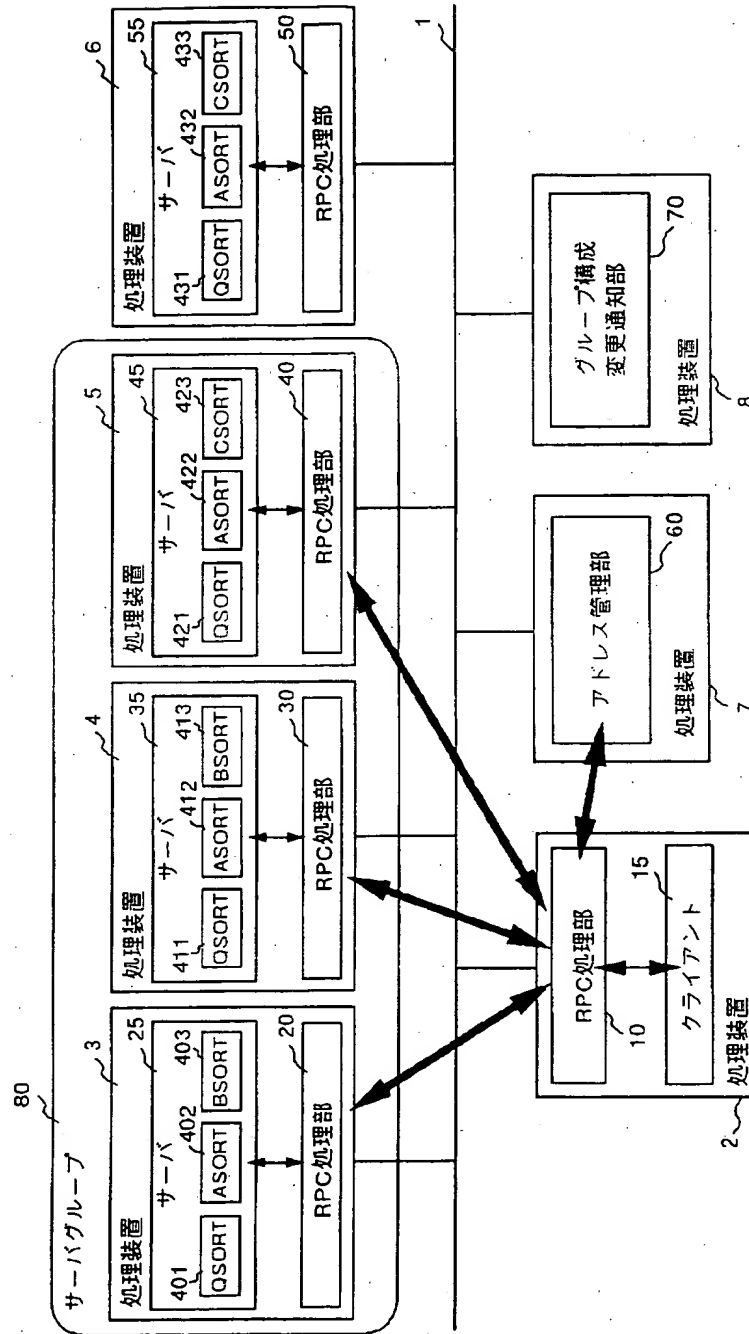
【図23】第3の実施例におけるRPC要求メッセージ及び応答メッセージを示すフォーマット図。

#### 【符号の説明】

1…通信媒体（ネットワーク）、2～8…処理装置（計算機）、10、20、30、43、50…RPC処理部、11…RPCクライアント処理部、12、22、32、42、52…RPCサーバ処理部、13…アドレス管理テーブル、14、24、34、44、54…グループ管理テーブル、143…グループ構成変更通番部、25、35、45、55、401、402、403、411、412、413、421、422、423、431、432、433…サーバ、60…アドレス管理部、70…グループ構成変更通知部、80…サーバグループ、100…サーバ変更通知RPCメッセージ、101…RPC制御ヘッダ部、102…RPC要求内容識別子部、103…変更サーバ処理識別子部、110…RPC要求メッセージ、111…クライアント要求データ部、120…RPC応答メッセージ、121…グループ構成変更通知部、122…サーバ実行結果データ部。

【図 1】

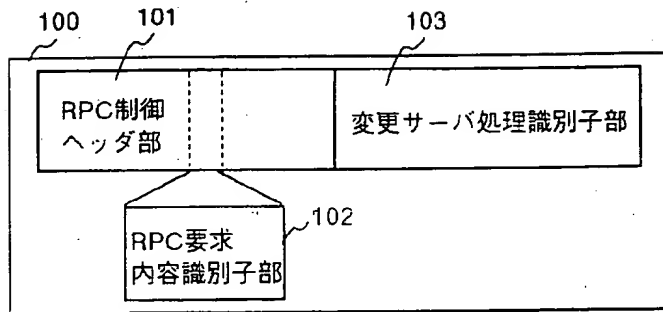
図 1



【図2】

図 2

サーバ変更の通知RPCメッセージ



【図6】

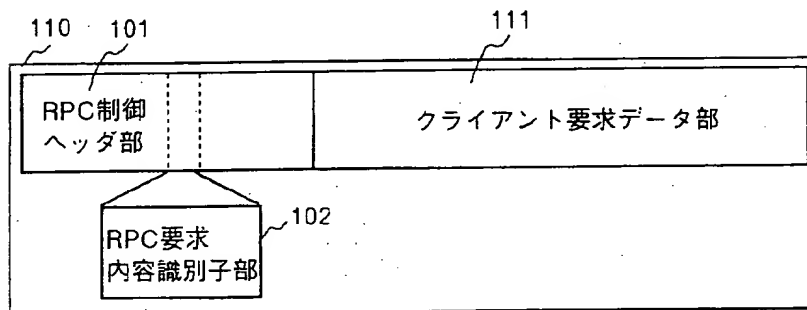
図 6

| 処理識別子部 | アドレス部                |
|--------|----------------------|
| QSORT  | 133.144.8.159 : 8080 |
| QSORT  | 133.144.8.160 : 8052 |
| QSORT  | 133.144.8.120 : 8031 |
| ASORT  | 133.144.9.120 : 6198 |
| ⋮      | ⋮                    |

【図3】

図 3

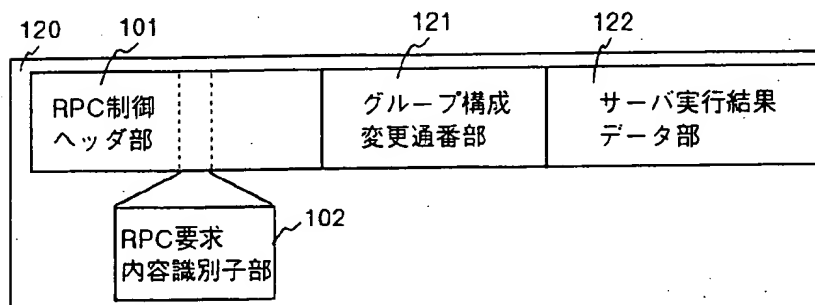
RPC要求メッセージ



【図4】

図 4

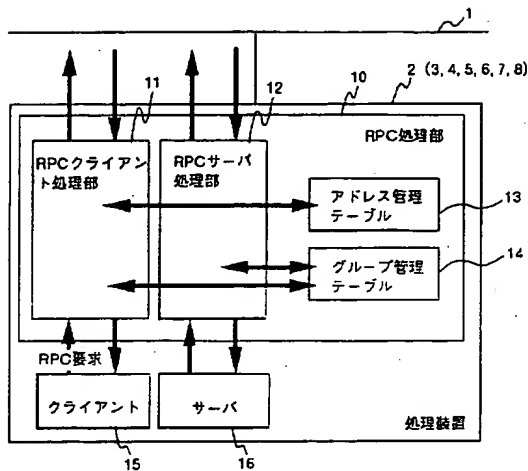
RPC応答メッセージ





【図 5】

図 5



【図 7】

図 7

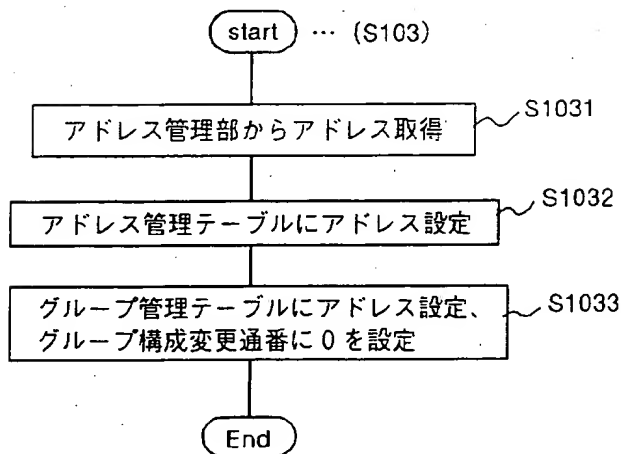
| 処理識別子部 | アドレス部              | グループ構成変更通番部 |
|--------|--------------------|-------------|
| QSORT  | 133.144.8.159:8080 | 2           |
| QSORT  | 133.144.8.160:8052 | 3           |
| QSORT  | 133.144.8.161:8031 | 5           |
| ASORT  | 133.144.9.120:6198 | 2           |
| ...    | ...                | ...         |

| 処理識別子部 | アドレス部              | グループ構成変更通番部 |
|--------|--------------------|-------------|
| QSORT  | 133.144.8.159:8080 | 2           |
| QSORT  | 133.144.8.160:8052 | 3           |
| QSORT  | 133.144.8.161:8031 | 6           |
| ASORT  | 133.144.9.120:6198 | 2           |
| ...    | ...                | ...         |

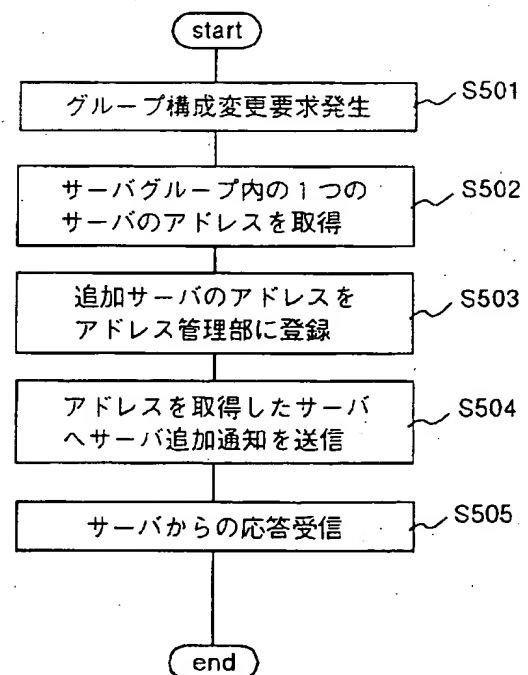
【図 9】

図 9



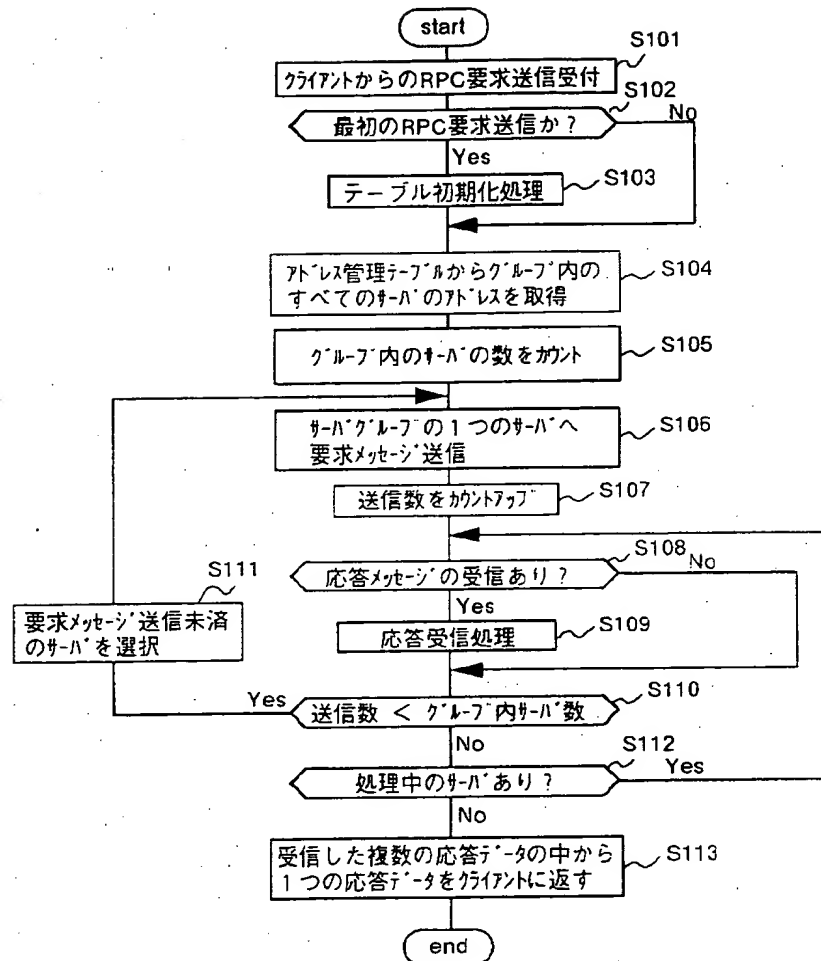
【図 1.3】

図 1 3



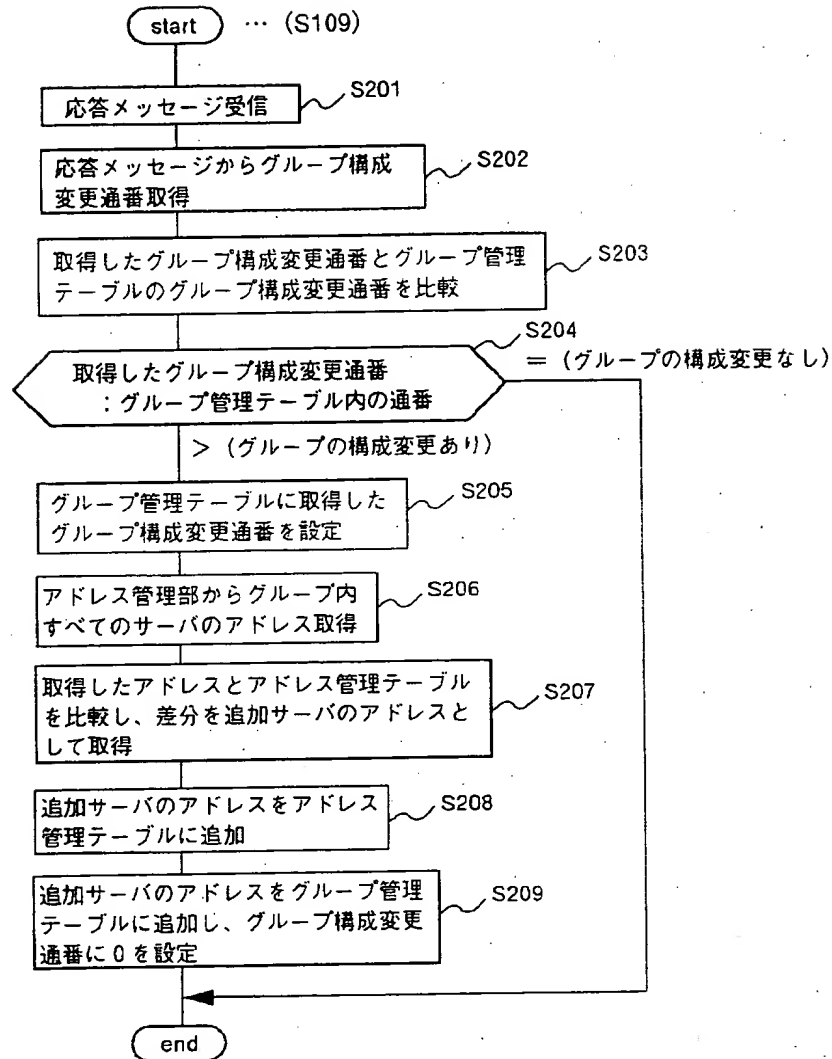
【図8】

図 8



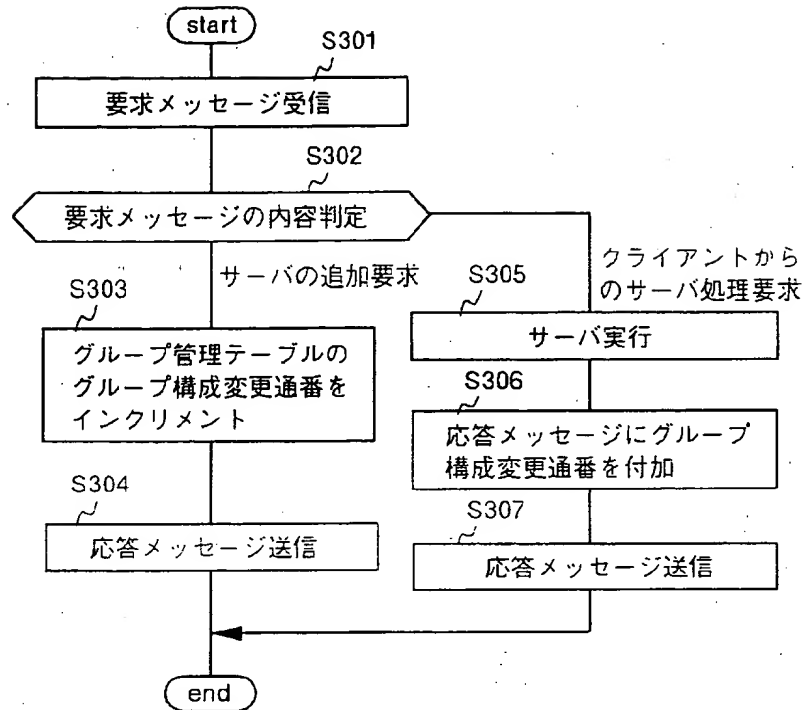
【図 10】

図 10



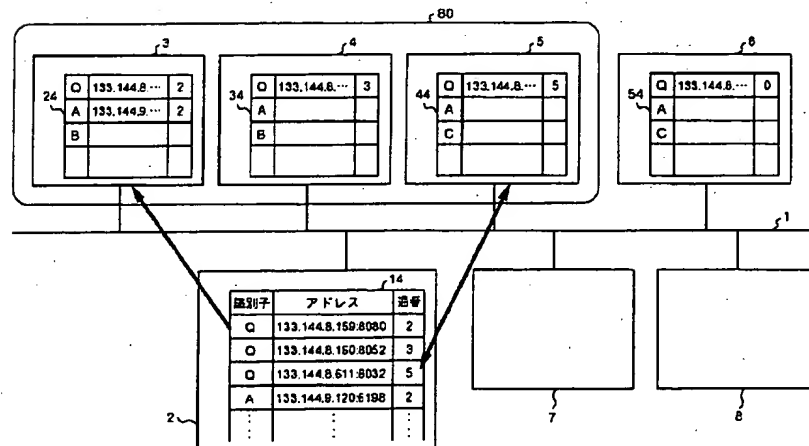
【図11】

図 11



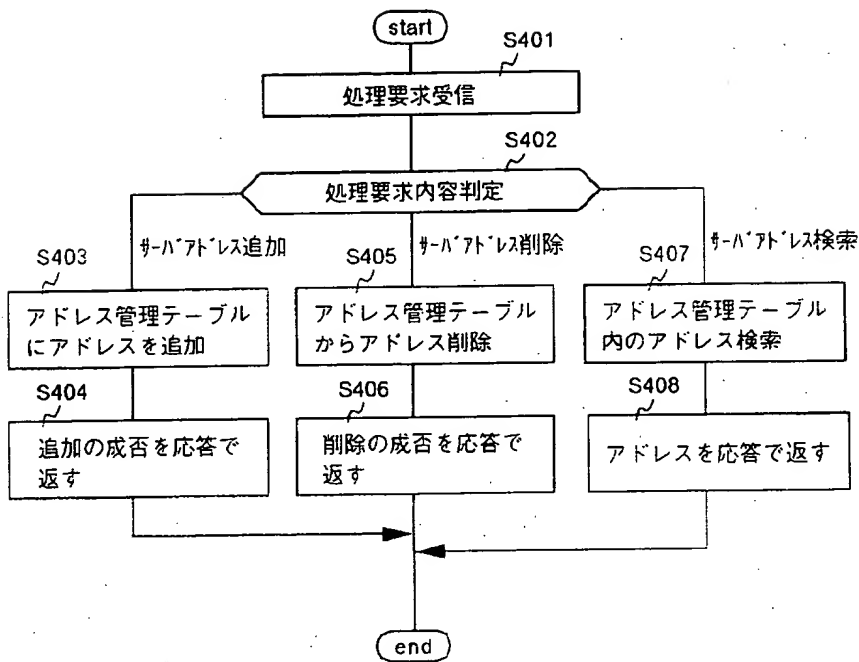
【図15】

図 15



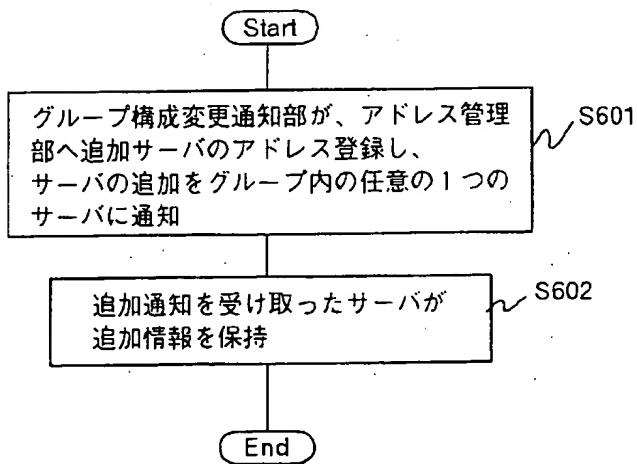
【図12】

図 1 2

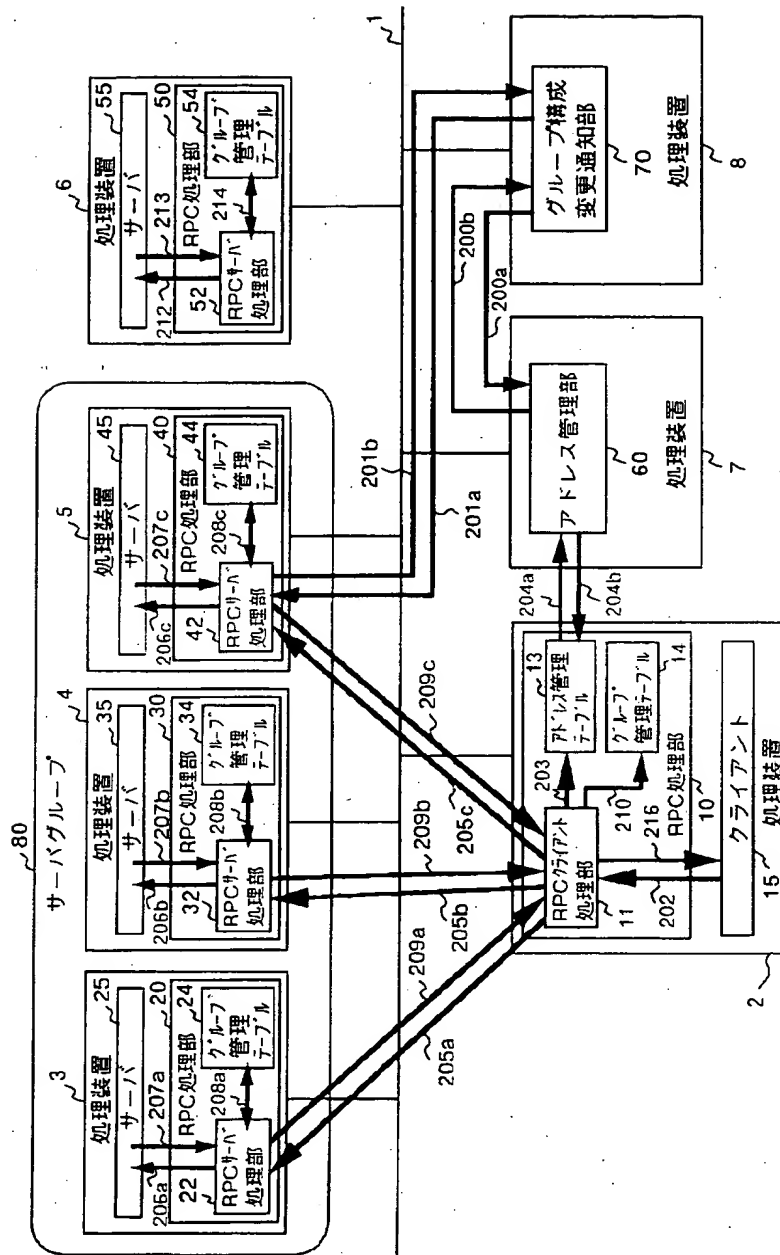


【図16】

図 1 6

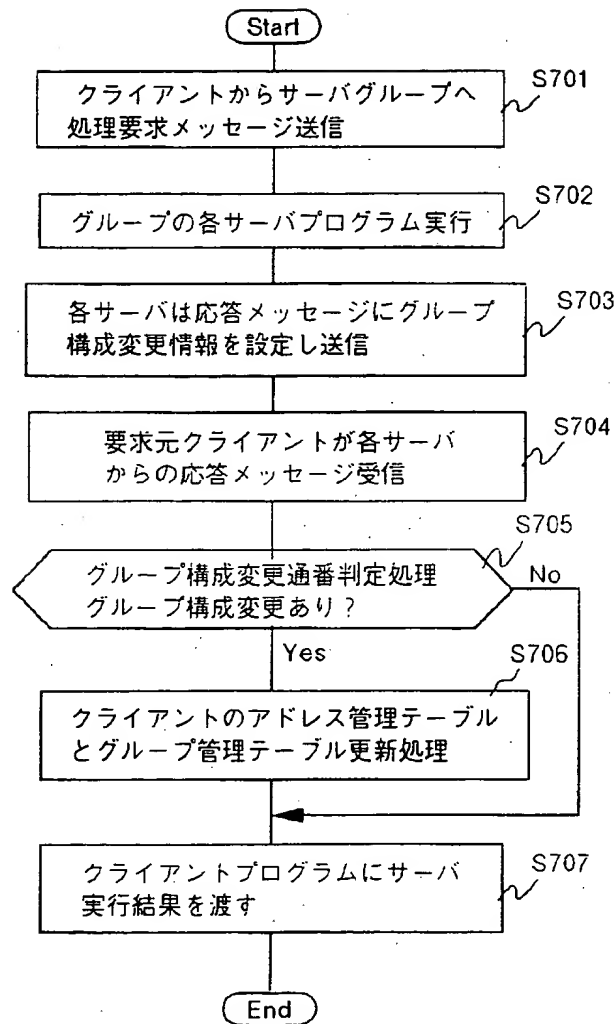


【図14】



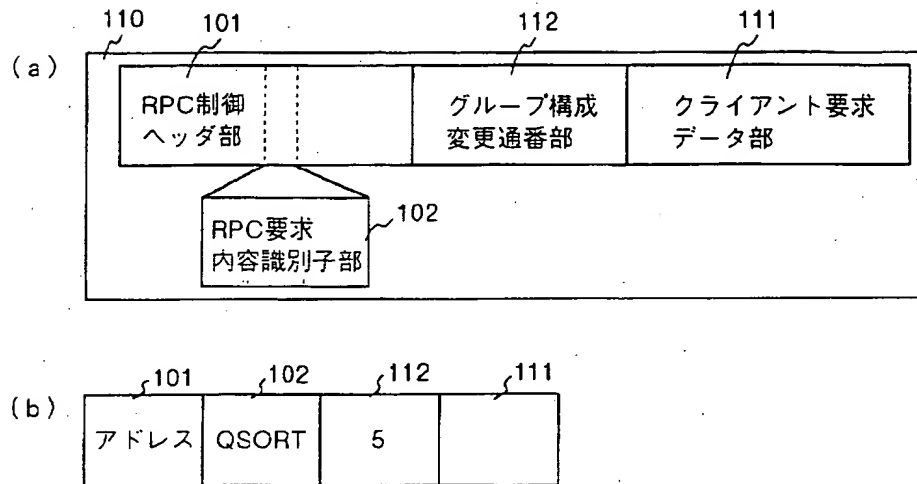
【図17】

図 17



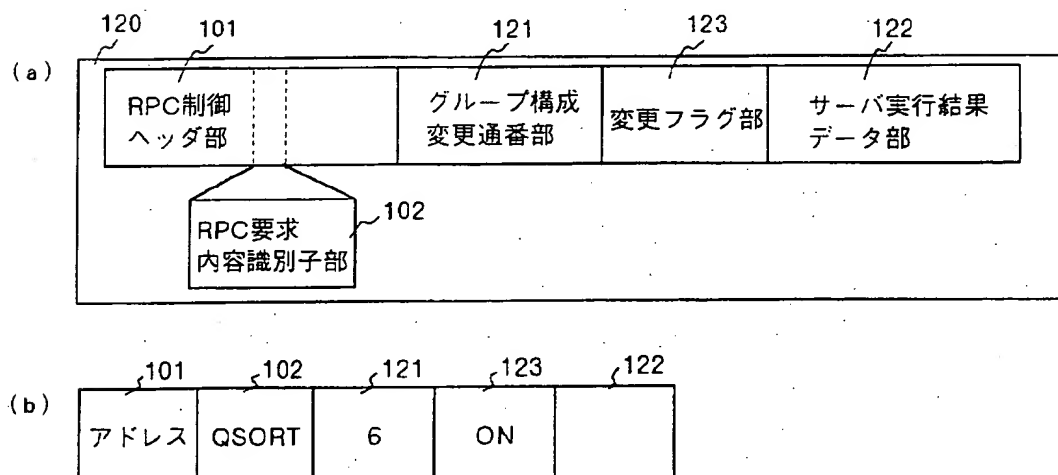
【図 18】

図 18



【図 19】

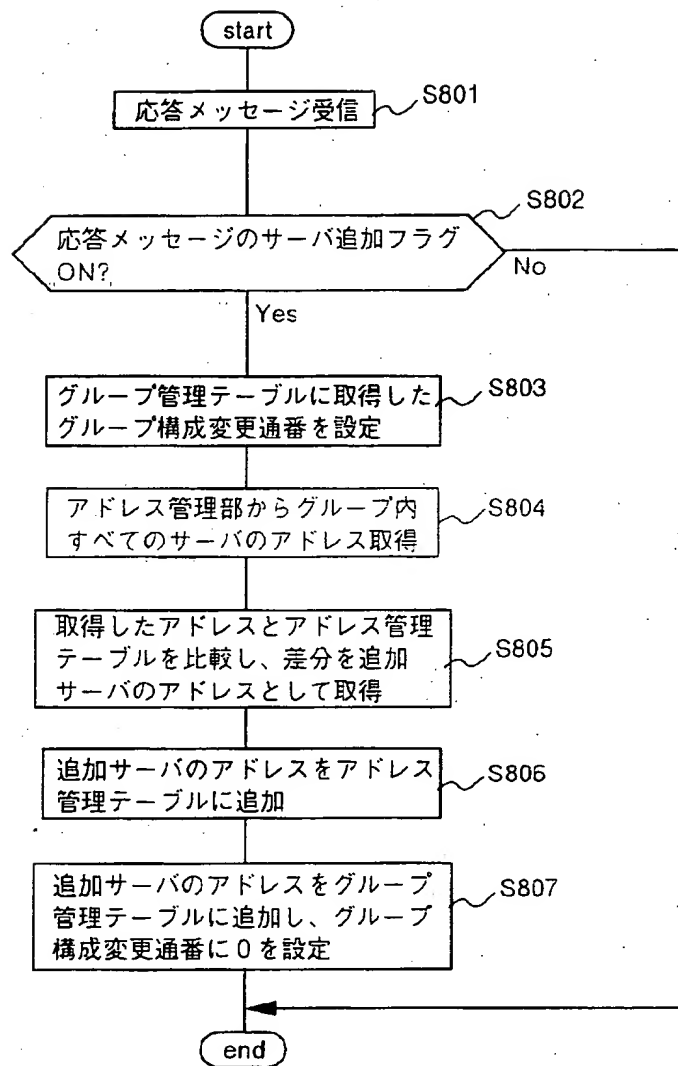
図 19





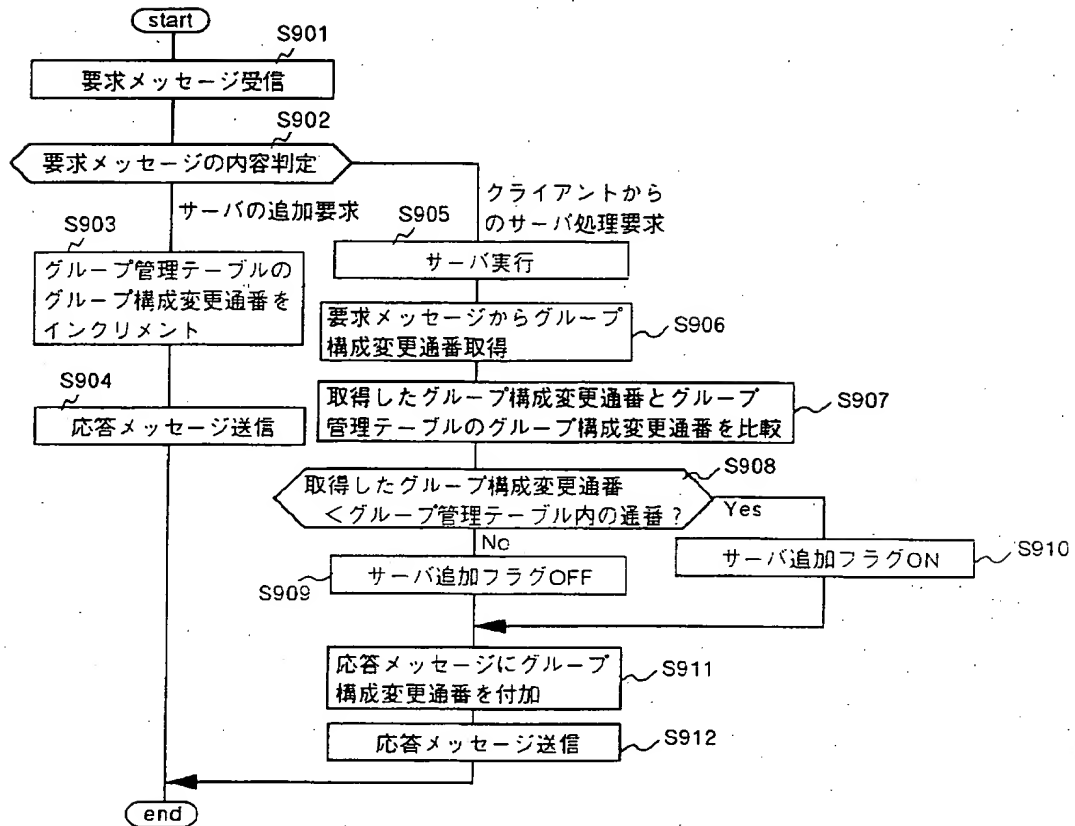
【図 20】

図 20

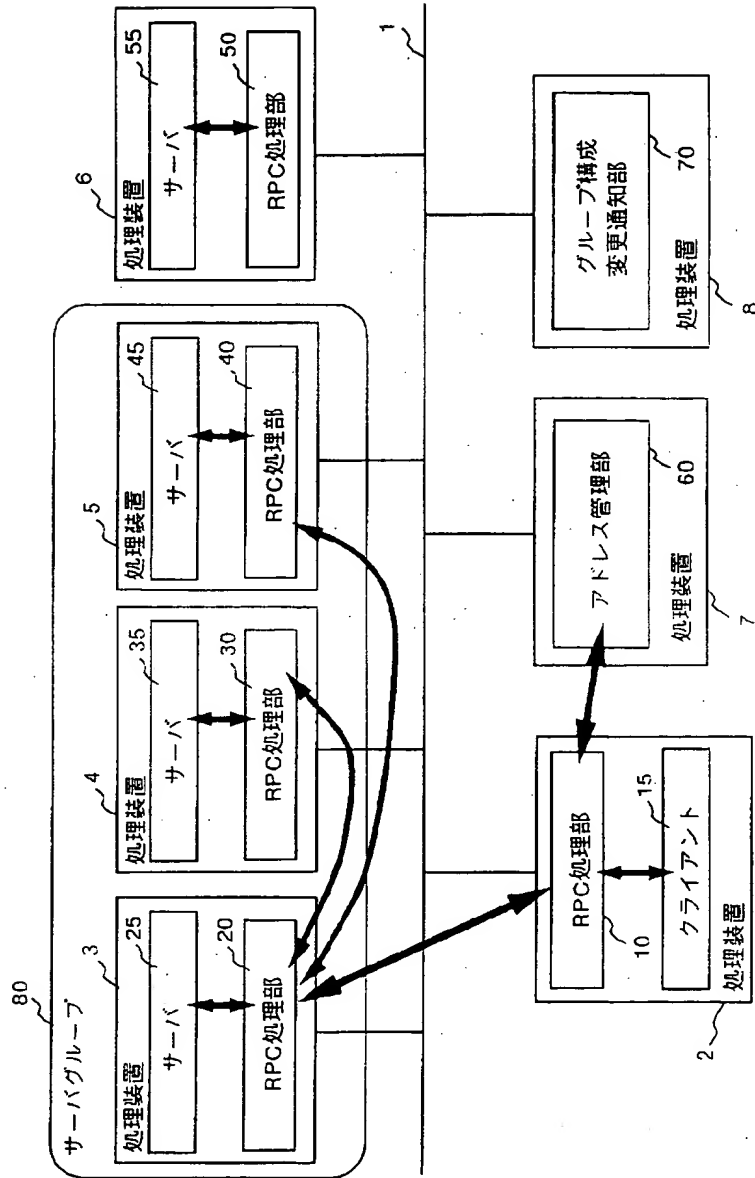


【図21】

図 21

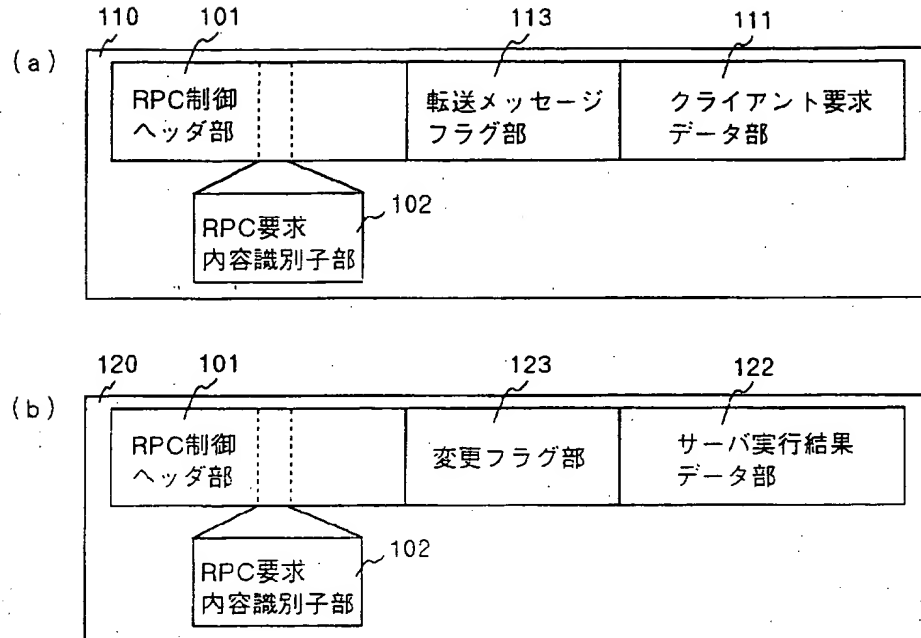


【図22】



【図 23】

図 23



フロントページの続き

(72)発明者 安東 宣善  
神奈川県川崎市麻生区王禅寺1099番地 株  
式会社日立製作所システム開発研究所内

(72)発明者 平澤 茂樹  
神奈川県川崎市麻生区王禅寺1099番地 株  
式会社日立製作所システム開発研究所内

(72)発明者 綿谷 洋  
茨城県日立市大みか町五丁目2番1号 株  
式会社日立製作所大みか工場内

(72)発明者 松崎 健二  
茨城県日立市大みか町五丁目2番1号 日  
立プロセスコンピュータエンジニアリング  
株式会社内

(72)発明者 田村 清一  
茨城県日立市大みか町五丁目2番1号 日  
立プロセスコンピュータエンジニアリング  
株式会社内

(72)発明者 鈴木 淳  
茨城県日立市大みか町五丁目2番1号 日  
立プロセスコンピュータエンジニアリング  
株式会社内